In the Specification

Please amend the specification of this application as follows:

Rewrite the paragraph at page 1, lines 8 to 9 as follows:

a1

--This application claims priority under 35 USC §119(e)(1) of Provisional Application No. 60/183,527, filed February 18, ~~2000 (TI-30302PS)~~ 2000.--

Rewrite the paragraph at page 4, lines 24 to 27 as follows:

a2

--In another embodiment of the invention, the final result is rounded at a mid-position and shifted to a bit length less then the bit length of the combined product. In another embodiment, the rounding value is ~~2**15,~~ 2^{15}, or 0x8000.--

Rewrite the paragraph at page 7, line 16 to page 8, line 14 as follows:

a3

--Figure 1 is a block diagram of a digital system with a digital signal processor (DSP), microprocessor 1, showing components thereof pertinent to an embodiment of the present invention. In microprocessor 1 there are shown a central processing unit (CPU) 10, data memory 22, program memory/cache 23, peripherals 60 and an external memory interface (EMIF) with a direct memory access (DMA) 61. CPU 10 further has an instruction fetch/decode unit 10a-c, a plurality of execution units, including an arithmetic and load/store unit D1, a multiplier M1, an ~~ALU/ shifter~~ ALU/shifter unit S1, an arithmetic logic unit ("ALU") L1, a shared multi-port register file 20a from which data are read and to which data are written. Instructions are fetched by fetch unit 10a from instruction memory 23 over a set of busses 41. Decoded instructions are provided from the instruction fetch/decode unit 10a-c to the functional units D1, M1, S1, and L1 over various sets of control lines which are not shown. Data are provided to/ from a3 the register file 20a from/to to load/store units D1 over a first set of busses 32a, to multiplier M1 over a second set of busses 34a, to ALU/shifter unit S1 over a third set of busses 36a and to ALU L1 over a fourth set of busses 38a. Data are provided to/from the memory 22 from/to the load/store units D1 via a fifth set of busses 40a. Note that the entire data path described above is duplicated with register file 20b and execution units D2, M2, S2, and L2. In this embodiment of the present invention, two unrelated aligned double word (64 bits) load/store transfers can be made in parallel between CPU 10 and data memory 22 on each clock cycle using bus set 40a and bus set 40b. A single non-aligned double word load/store transfer is performed by scheduling a first .D unit resource and two load/store ports on a target memory. Advantageously, a second .D unit can perform 32-bit logical or arithmetic instructions in addition to the .S and .L units while the address port of the second .D unit is being used to transmit one of two contiguous addresses provided by the first .D unit.--

Rewrite the paragraph at page 8, lines 22 to 29 as follows:

a4 --Note that the memory 22 and memory 23 are shown in Figure ~~19~~ 1 to be a part of a microprocessor 1 integrated circuit, the extent of which is represented by the box 42. The memories 22-23 could just as well be external to the microprocessor 1 integrated circuit 42, or part of it could reside on the integrated circuit 42 and part of it be external to the integrated circuit 42. These are matters of design choice. Also, the particular selection and number of execution units are a matter of design choice, and are not critical to the invention.--

Rewrite the paragraph at page 9, lines 11 to 18 as follows:

a5 --A detailed description of various architectural features of the microprocessor 1 of Figure 1 is provided in co-assigned U.S.

a5 Patent ~~application S.N. 09/012,813 (TI-25311)~~ No. 6,182,203 and is incorporated herein by reference. A description of enhanced architectural features and an extended instruction set not described herein for CPU 10 is provided in co-assigned U.S. Provisional Patent application S.N. 60/183,527 ~~(TI-30302,~~ now U.S. Patent Application Serial No. 09/703,096 entitled *Microprocessor with Improved Instruction Set Architecture*~~)~~ and is incorporated herein by reference.--

Rewrite the paragraph at page 10, lines 6 to 8 as follows:

a6 --There are 32 valid register pairs for 40-bit and 64-bit data, as shown in Table ~~4~~ 1. In assembly language syntax, a colon between the register names denotes the register pairs and the odd numbered register is specified first.--

Rewrite the paragraph at page 10, line 10 as follows:

a7 --Table ~~4~~ 1. 40-Bit/64-Bit Register Pairs--

Rewrite the paragraph at page 10, lines 17 to 19 as follows:

a8 --Referring again to Figure 2, the eight functional units in processor 10's data paths can be divided into two groups of four; each functional unit in one data path is almost identical to the corresponding unit in the other data path. The functional units are described in Table ~~5.~~ 2.--

Rewrite the paragraph at page 11, line 3 as follows:

a9 --Table ~~5.~~ 2. Functional Units and Operations Performed--

Rewrite the paragraph at page 12, lines 13 to 19 as follows:

a10 --Table ~~6~~ 3 defines the mapping between instructions and functional units for a set of basic instructions included in the present embodiment. Table ~~7~~ 4 defines a mapping between instructions and functional units for a set of extended instructions in an embodiment of the present invention. Alternative embodiments of the present invention may have different sets of instructions and functional unit mapping. Tables ~~6~~ 3 and ~~7~~ 4 are illustrative and are not exhaustive or intended to limit various embodiments of the present invention.--

a10

Rewrite the paragraph at page 13, line 1 as follows:

a11

--Table ~~6.~~ 3. Instruction to Functional Unit Mapping of Basic Instructions--

Rewrite the paragraph at page 14, lines 1 and 2 as follows:

a12

--Table ~~7.~~ 4, Instruction to Functional Unit Mapping of Extended Instructions--

Rewrite the paragraph at page 15, lines 21 to 27 as follows:

a13

--The pipeline operation, from a functional point of view, is based on CPU cycles. A CPU cycle is the period during which a particular execute packet is in a particular pipeline stage. CPU cycle boundaries always occur at clock cycle boundaries; however, memory stalls can cause CPU cycles to extend over multiple clock cycles. To understand the machine state at CPU cycle boundaries, one must be concerned only with the execution phases (E1-E5) of the pipeline. The phases of the pipeline are described in Table ~~8.~~ 5.--

Rewrite the paragraph at page 16, line 1 as follows:

a14

--Table ~~8.~~ 5. Pipeline Phase Description--

Rewrite the paragraph at page 17, line 23 to page 18, line 5 as follows:

a15

--In step 310, a first pair of elements are multiplied together to form a first ~~product~~ product. The most significant 16-bit value of the first source operand and the most significant 16-bit value of the second source operand are multiplied together to form a 32-bit first product. In step 311, a second pair of elements are multiplied together to form a second product. The least significant 16-bit value of the first source operand and the least significant 16-bit value of the second source operand are multiplied together to form a 32-bit second product. The two products are formed simultaneously by a pair of multiplier circuits in the M1 functional unit during the E1 execute phase. In this embodiment, one of the 16-bit values of each pair of elements is treated as a signed number and the other 16-bit value of each pair of elements is treated as an unsigned number. Each product is treated as a signed integer value.--

Rewrite the paragraph at page 20, lines 1 to 3 as follows:

a16

--Referring still to Figure 3, the present embodiment defines several rounding dot product instructions that are specified by the OP field, as described in Table ~~9,~~ 6, while several examples are provided in Table ~~10.~~ 7.--

Rewrite the paragraph at page 20, line 5 as follows:

a17

--Table ~~9.~~ 6. Rounding Dot Product Instructions--

Rewrite the paragraph at page 20, line 10 as follows:

a18

--Table ~~10.~~ 7. Rounding Dot Product Examples--

Rewrite the paragraph at page 23, lines 24 to 29 as follows:

a19

--Galois field multiply unit 460 performs Galois multiply in parallel with multiplier mpy0, mpy1. For output from the M unit, the Galois multiply result is muxed with the multiply result.

a19 Details of the Galois multiply unit are provided in co-assigned U.S. Patent application ~~S.N. ________ (TI-26013)~~ Serial No. 09/507,187 to David Hoyle entitled *Galois Field Multiply* and is incorporated herein by reference.--

Rewrite the paragraph at page 27, lines 12 to 15 as follows:

a20 --Additional information on embodiments of paired multiplier circuits is provided in co-assigned U.S. Patent application ~~S.N. ________ (TI-26010)~~ Serial No. 09/703,093 to David Hoyle entitled *Data Processor with Flexible Multiply Unit* and is incorporated herein by reference.--

Rewrite the paragraph at page 30, lines 4 to 11 as follows:

a21 --Within an M unit, various combinations of fixed and/or variable shifters can be provided. Other mid-point rounding locations may be selected such that the rounding value is ~~2**n~~ 2^n and the intermediate result is shifted n+1. For example a rounding value of ~~2**11~~ 2^{11} with a twelve bit right shift. Alternatively, instead of performing a right shift of n+1, a left shift can be performed to shift the final result to a more significant portion of a 64-bit output register, for example, to form a final result such that the n lsbs of the intermediate result stored in a destination register are truncated.--